A Novel Crossfeed Audio Plugin For An Improved Headphone Listening Experience

Helmut Keller

This paper describes the theory and implementation of a novel audio plugin designed to enhance the listening experience when using headphones. Listening through headphones lacks the natural crosstalk between the listener's ears that occurs with a standard loudspeaker setup. In the plugin, the interaural transfer function at an azimuth angle of 30° and an elevation angle of 0° is approximated using a combination of a second-order minimum phase filter, and a delay line. This approximated transfer function is implemented in the crossfeed path of the plugin. As a result, the spatial perception is more realistic compared to existing solutions. Additionally, a novel equalizer allows for a seamless transition between full mono compatibility and minimal sound color changes for typical stereophonic signals.

The plugin is available for free on the author's website. It has the JSFX plugin format and operates natively in the digital audio workstation Reaper. Furthermore, with the assistance of the YSFX bridge plugin released by Joep Vanlier, it can run on nearly every audio plugin host. YSFX is also available for free as a VST3, AU, or CLAP plugin for Windows, Mac OS, and Linux.

1. Theory

The European Broadcasting Union recommends placing the loudspeakers of a stereo system at azimuth angles a of -30° for the left speaker and +30° for the right speaker, directly in front of the listener. The recommended elevation angle e is 0°, and the ideal distance r from the listener is between 2 to 4 meters [1].

The sound from the two speakers reaches both ears, but with different head-related transfer functions (HRTFs). If we disregard any room reflections and assume symmetrical heads, we can derive:

$$H_{\text{direct}}(f)=HRTF_{\text{left ear}}(a_{\text{left}},e,r,f)=HRTF_{\text{right ear}}(a_{\text{right}},e,r,f) \quad (1)$$

$$H_{\text{cross}}(f)=HRTF_{\text{left ear}}(a_{\text{right}},e,r,f)=HRTF_{\text{right ear}}(a_{\text{left}},e,r,f) \quad (2)$$

$$H_{\text{inter}}(f)=\frac{H_{\text{cross}}(f)}{H_{\text{direct}}(f)} \quad (3)$$

The interaural transfer function, H_{inter}, is essential for the spatial perception, while H_{direct} can be viewed as a component of an equalizer that influences the color of the perceived sound. If we aim to emulate the HRTFs in a crossfeed circuit for headphones, as demonstrated in **Fig. 1**, we need to implement H_{inter} as realistically as possible. However, H_{direct} should be selected in such a way that mono compatibility is primarily maintained. This means that a monophonic signal should not be significantly altered by the crossfeed circuit. The rationale behind this approach is that good headphones are assumed to be already equalized such that monophonic signals sound natural.

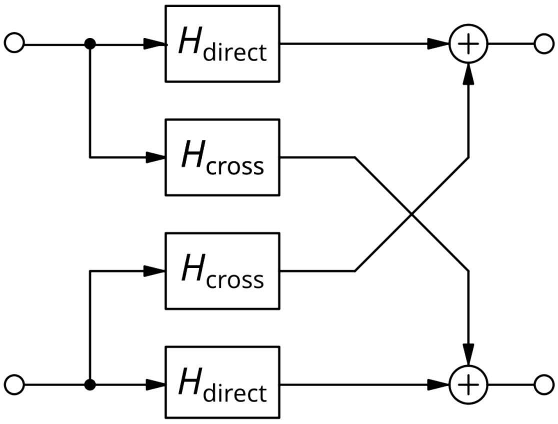

Fig.1: Crossfeed circuit for headphones

The transfer functions for the monophonic and the side signals are:

$$H_{mono}(f) = H_{direct}(f) + H_{cross}(f) \tag{4}$$

$$H_{side}(f) = H_{direct}(f) - H_{cross}(f) \tag{5}$$

A conducive redefinition is:

$$H_{direct}(f) = \frac{1}{1 + k \cdot H_{inter}(f)} \tag{6}$$

$$H_{cross}(f) = \frac{H_{inter}(f)}{1 + k \cdot H_{inter}(f)} \tag{7}$$

This redefinition of H_{direct} and H_{cross} retains the original H_{inter} and allows us to smoothly transition between no change in the direct signal and full mono compatibility. A blending parameter $k = 0$ yields a flat H_{direct}, while $k = 1$ produces a flat H_{mono}.

We may also define a transfer function H_{ind} based on independent left and right signals. We may assume that the amplitude spectra of the left and right input signals are the same while the phase spectra differ randomly. Two uncorrelated pink noise signals serve as an example of such a signal pair.

$$H_{ind} = \sqrt{|H_{direct}|^2 + |H_{cross}|^2} \tag{8}$$

Since stereo recordings contain a certain amount of independent audio signals on the left and right channels, the blending parameter k should be set such that both H_{mono} and H_{ind} are reasonably flat.

2. Some milestones in the history of crossfeed circuits

Benjamin B. Bauer proposed the first crossfeed circuit for headphones. He was granted a US patent [2] for it in 1963. Bauer attempted to approximate the HRTFs for azimuth angles of +/- 45°. He used minimum phase filters in his method, resulting in unrealistic interaural delay times at high frequencies. The magnitude of the interaural transfer function was too low for frequencies exceeding 4 kHz. The circuit was completely mono compatible.

Sigfried Linkwitz [3] designed a substantially simpler circuit in 1971. For the crossfeed path, a first order lowpass filter with a low frequency gain of -5 dB and a pole frequency of 700 Hz was employed, whereas the direct path used a first order lowshelf filter with a low frequency gain of -2 dB. His circuit had the same issues as Bauer's circuit, but the magnitude of the interaural transfer function was too small, even at low frequencies. The circuit was not mono compatible.

Around 2005, Boris Mikhaylow released the VST plugin BS2B [4]. He used the same circuit as Linkwitz, but adjusted the parameters. He used lowpass gains of -6.75, -8.0, and -10.92 dB in his three settings (1, 2, and 3). The lowpass filter had pole frequencies of 700, 700, and 650 Hz. The low frequency gains of the low-shelf filters were -2.25, -2.0, and -1.42 dB. The low shelf filters had pole frequencies of 1038.8, 975.0, and 893.6 Hz. None of his settings came close to the original Bauer circuit because he used Linkwitz's reduced circuit.

The first setting was devised by Boris Mikhaylow himself. He claimed that this setting would most closely resemble a speaker setup with azimuth angles of +/- 30° and a distance of 3 meters. The second setting was inspired by Chu Moy [5], while the third was influenced by Jan Meier [6]. All three configurations were less realistic than the original Linkwitz circuit because the magnitude of the interaural transfer function was reduced at all frequencies.

Some audio interfaces manufactured by RME provide a crossfeed function. Page 19 of the ADI-2 DAC FS user manual [7] describes this feature. The RME presets 4, 3, and 2 correspond to the original BS2B options 1, 2, and 3. The presets 1 and 5 are modifications by RME.

The original BS2B VST plugin is not supported by 64-bit Windows. However, the company Resonic issued the free 64-bit VST3 plugin BS2BR [8], which included the original three options as well as an additional Resonic preset. N.B.: this plugin cuts signals over 0 dBFS and has a reduced total gain.

A universal crossfeed plugin was proposed by Francis F. Li in 2015 [9]. The idea to implement H_{inter} as a series connection of a delay line and a minimum phase filter was a first. Unfortunately, Li replaced a realistic H_{direct} with a unity gain block and utilized a realistic H_{cross} as H_{inter}. As a result of this simplification, the magnitude of the effective H_{inter} was too small at low frequencies and too high at high frequencies. This is unfortunate because Li already utilized high-order filters which are likely to more easily produce highly realistic results.

3. Approximation of the interaural transfer function

The author analyzed publicly available HRTF measurements of an artificial head [10] and of 48 subjects [11] at azimuth angles of +/- 30° and an elevation angle of 0°. The source distance r was

1.4 m for the artificial head and 1.5 m for the 48 individuals. Both distances are large enough to assume far-field conditions; the HRTFs are therefore also applicable for higher distances. The author calculated two H_{inter} for two distinct pinnae sizes of the artificial head, as well as all 96 H_{inter} of the 48 participants in the above-mentioned measurements. For both situations, the mean gain and phase delay were determined. **Fig. 2** displays both results (bold black lines for the 48 subjects and thin black lines for the artificial head).

In **Fig. 2**, a bold red line represents the author's approximation using a series connection of a fractional delay line and a second order minimum phase filter. The other colored thin lines represent Linkwitz and Mikhaylov's estimations.

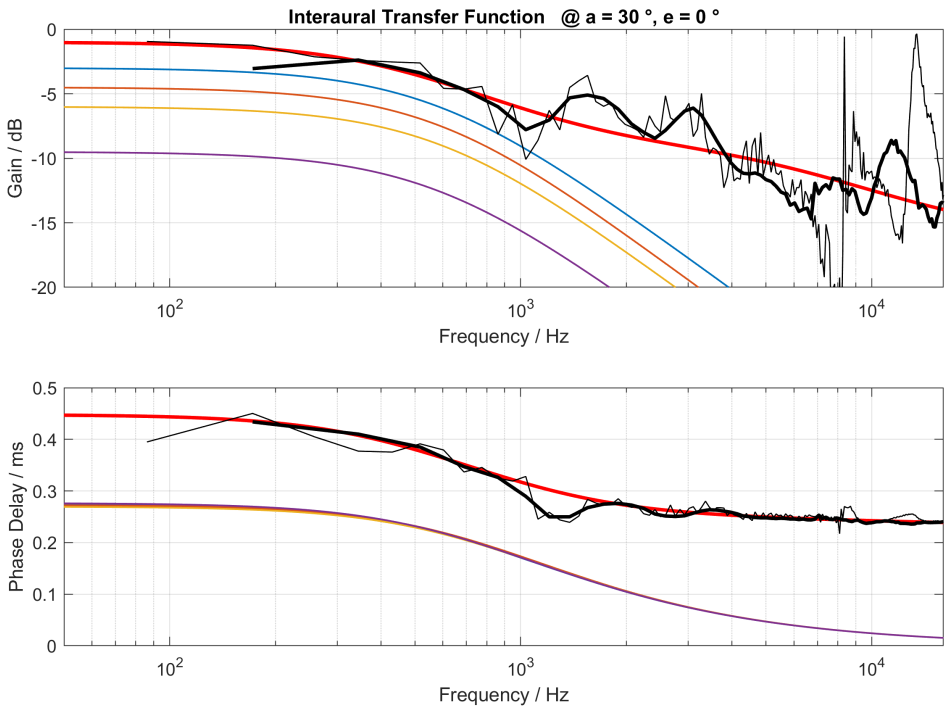

Fig.2: Measured interaural transfer functions and some approximations.

The author's gain-approximation is remarkably accurate at low frequencies and deviates only a few dB at middle and high frequencies. The author's phase-delay-approximation is fairly good at all frequencies except for a minor hump around 1.25 kHz. Of course, higher order filters could perform better, but the variations between individual interaural transfer functions are already on the same order of magnitude as the deviations in the approximation. Thus, the author does not expect a significant improvement in the spatial perception by employing higher order filters unless the individual HRTFs of the actual listener are known and may be considered.

The gain-approximation of the Linkwitz approach is better than the one by Mikhaylov, but it is still too low at low frequencies and much too low at high frequencies. The phase delay approximations by both Linkwitz and Mikhaylov are nearly equivalent, but significantly too low for all frequencies.

4. Implementation of the plugin

A second-order minimum phase filter connected in series with a fractional delay line approximates the interaural transfer function.

The second-order minimum-phase filter is based on an analog prototype of a universal tone control circuit (tone stack). The filter was initially defined by the author in [12]:

$$H_{\mathrm{TST}}(f)=\frac{B+M\cdot\frac{j\cdot f}{Q\cdot f_0}-T\cdot\frac{f^2}{f_0^2}}{1+\frac{j\cdot f}{Q\cdot f_0}-\frac{f^2}{f_0^2}} \tag{9}$$

The parameters *B*, *M*, and *T* control the bass, middle, and treble levels, respectively. The parameter f_0 represents the mid-band's center frequency, and the reciprocal of *Q* determines its relative bandwidth. All parameter values should be non-negative real numbers. Thus, this filter is also the most general second-order minimum phase filter. The approximation in **Fig. 2** was accomplished using the following values: B = 0.8915, M = 0.3448, T = 0.1585, Q = 0.25, and f_0= 1.8 kHz.

The analog prototype filter is converted to a direct-form-2 digital filter using the bilinear transformation with f_0 and *Q* prewarping, as described in [12].

The fractional delay line with a delay time t_d is implemented as a series connection of a delay line with an integer number of sample intervals *T* and a first-order digital all-pass filter for the fractional part of the delay time. The delay time of the all-pass filter ranges from 0.618034·*T* to 1.618034·*T*. This minimizes the all-pass filter's time constant as well as phase deviations from an ideal delay line. The approximation in **Fig. 2** was attained with a delay time of 235 µs.

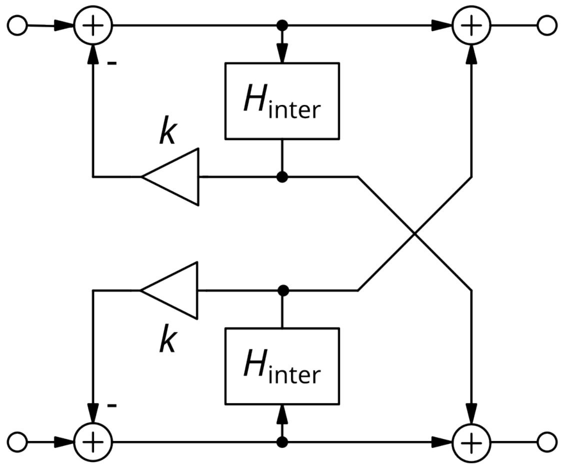

Fig. 3: Modified crossfeed circuit

Fig. 3 depicts the modified crossfeed circuit, which employs the approximated transfer function H_{inter}. This is an accurate and efficient implementation of the redefined transfer functions H_{direct} and H_{cross} from equations (6) and (7). The circuit is feasible as long as t_d is bigger than 1.619034·*T*. The trick to avoid delay free loops is to separate H_{inter}'s fractional delay line into two parts: a delay of one sample interval T and a fractional delay line with a delay time of $t_d - T$.

The pseudo code below is an accurate implementation of the redesigned crossfeed circuit from **Fig. 3**. It must run for each new pair of left and right samples and overwrites them with the processed values.

```
direct_left = left – k * cross_left
direct_right = right – k * cross_right

left = direct_left + cross_right
right = direct_right + cross_left

cross_left = tone_stack(B, M, T, f0, Q, direct_left)
cross_left = delay(td - T, cross_left)

cross_right = tone_stack(B, M, T, f0, Q, direct_right)
cross_right = delay(td - T, cross_right)
```

The user of the plugin can set a "Mono Compatibilty" parameter between zero and 100%. This corresponds to values of k ranging from 0 to 1.

The plugin is a JSFX plugin created in the Jesusonic scripting language. It runs natively in the digital audio workstation Reaper. It is compatible with almost every audio plugin host thanks to Joep Vanlier's bridge plugin YSFX. YSFX is available for free on [13] as a VST3, AU, or CLAP plugin for Windows, Mac OS, and Linux.

The author's plugin may be downloaded for free from [14]. The source code is open and free to use, as long as the author's work is properly credited. However, any commercial use of the source code, or of the main ideas from this work requires the author's express consent. The author will grant permission to any serious audio company without any expectation of license fees.

Fig.4: The GUI of the author’s croosfeed plugin.

5. Derived transfer functions and listening tests

The gain plots of some derived transfer functions of the authors plugin are shown in **Fig. 5**. The color code for the mono-compatibility setting is: **0 %**, **60 %** and**100 %**.

The downside of a flat H_{mono} is obvious: the other transfer functions are not flat. This is not a fault of this plugin; it is simply the result of implementing realistic interaural transfer functions. The signals received by both ears influence the perceived sound color in a complicated way. Thus, none of the derived transfer functions represent the perceived sound color of stereophonic signals. Only for monophonic signals, there is no doubt that a flat H_{mono} will not change the perceived sound color when the plugin is enabled.

When listening to common stereophonic recordings, the author suggests using a mono compatibility value of approximately 60%. The perceived sound color change that may occur when the crossfeed plugin is engaged will be minimal in this typical application.

When the crossfeed plugin is enabled in any of its settings, the spacious impression improves uniformly and significantly. Especially hard panned signals sound much more natural. In general, the signals are no longer perceived to be located only between the two ears, but rather on a 60-degree-wide arc directly in front of the listener's head. Unfortunately, the perceived radius of this arc is in the same order of magnitude as the radius of a typical human head, rather than several meters as desired. This effect will depend on the individual listener and how well the individual HRTF's are matched by the plugin.

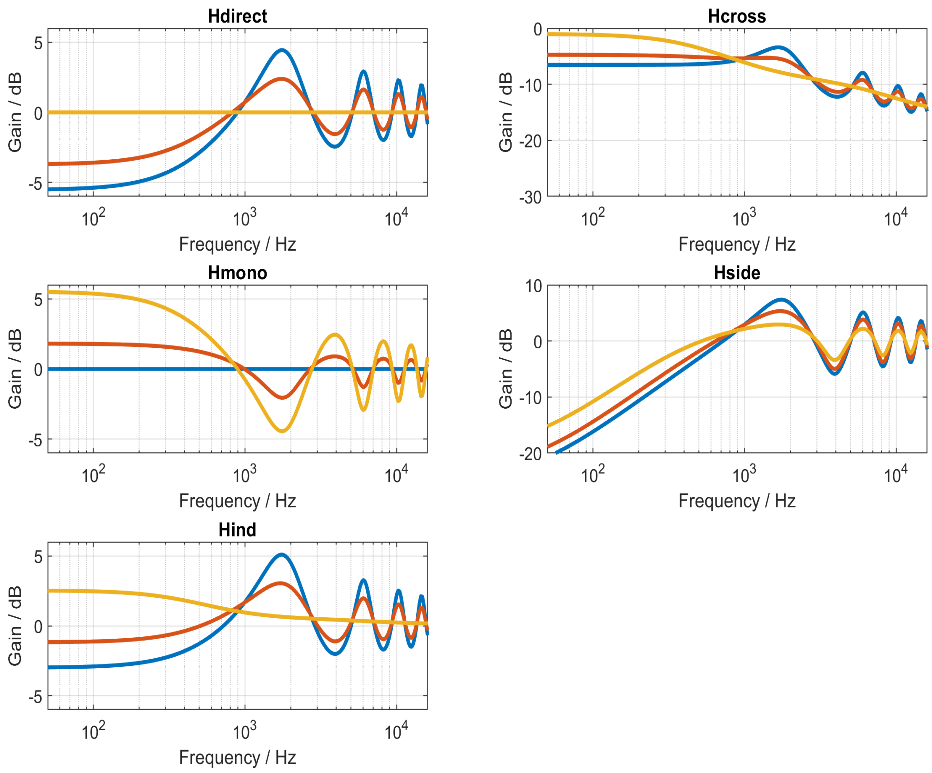

Fig. 5: Derived transfer functions of the authors plugin.

Fig. 6 displays the gain plots of the transfer functions for the Linkwitz- and Mikhaylov-circuits. The color of the Linkwitz-circuit is blue.

Mikhaylov's settings improve the spatial perception slightly, but not as significantly as the author's plugin. The first setting has the most impact. The Linkwitz-circuit was not available for hearing tests.

The author's website [15] has a web player for an audio file including signal samples that have been unprocessed (O), processed by the BS2BR plugin in the first Mikhaylov-setting (B), and processed by the author's plugin with 60% mono compatibility (K). The BS2BR plugin's gain loss has been corrected, and it is operating below its clipping threshold. For each signal sample, you will hear the following sequence: O-B-K.

The first signal is pink noise that pans continually from left to right. The second signal is composed of two independent pink noise signals: a mid-signal and a side-signal with a 6 dB lower strength. This signal approximates the characteristics of a normal stereo recording. The final three signals are brief snippets from popular stereo records.

The reader is invited to check out these demos!

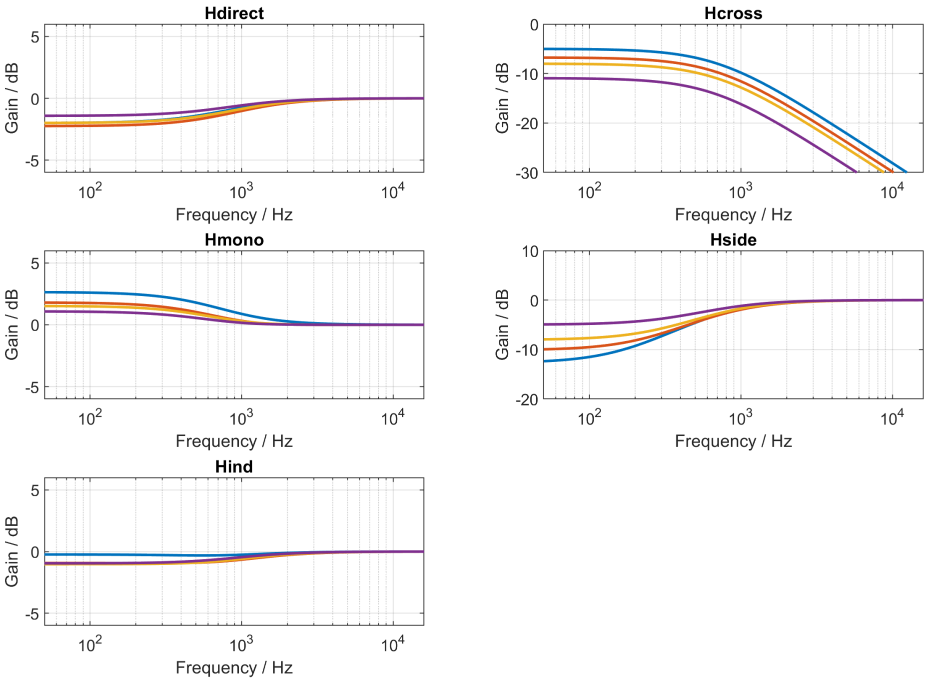

Fig.6: Transfer functions of the Linkwitz and Mikhaylov crossfeeders.

5. Summary

An innovative crossfeed audio plugin has been created to improve the listening experience with headphones. This solution offers a more realistic spatial perception than existing options. When a novel equalizer is applied to stereophonic signals, it produces only small sound color changes. There are no sound color changes for monophonic signals when the equalization is set to full mono compatibility. The free and open-source JSFX plugin is compatible with practically any audio plugin host thanks to Joep Vanlier's free bridge plugin YSFX.

The author's method produces a stereo image close to a 60 degrees-wide arc in front of the listener's head, with a perceived distance of about 10 cm. Individualized HRTFs which are moreover continuously modified via head tracking are most likely required for a more realistic distance perception. The tactile reaction to low frequency sound at high SPL-values is not implemented since this would necessitate the deployment of shakers. The acoustics of the listening space are not mimicked, either. This could be accomplished utilizing an ambience-simulation, such as a convolution reverb with impulse responses recorded with an artificial head in high-quality listening rooms or mastering studios.

6. Links

[1] EBU Tech 3276, 2nd edition, May1998:
https://tech.ebu.ch/docs/tech/tech3276.pdf

[2] Benjamin. B. Bauer: Patent US 30888997, 1963
https://patentimages.storage.googleapis.com/b2/fb/db/59b70b5e5cd1a2/US3088997.pdf

[3] Siegried Linkwitz: “Improved Headphone Listening”, Audio 1971
https://www.linkwitzlab.com/headphone-xfeed.htm

[4] Boris Mikhaylov: “Bauer stereophonic-to-binaural DSP.”
https://bs2b.sourceforge.net/

[5] Mu Choy: “An Acoustic Simulator For Headphone Amplifiers”
https://headwizememorial.wordpress.com/2018/03/09/an-acoustic-simulator-for-headphone-amplifiers/

[6]….Jan Meier: “An Enhanced-Bass Natural Crossfeed Filter”
https://headwizememorial.wordpress.com/2018/03/09/an-enhanced-bass-natural-crossfeed-filter/

[7] RME: User manual of ADI-DAC FS
https://rme-audio.de/downloads/adi2dacr_d.pdf

[8] Resonic: Download link for the BS2BR plugin
https://resonic.at/tools/bs2br

[9] Francis F. Li: “Improving headphone user experience in ubiquitous multimedia content consumption: A universal cross-feed filter”, IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, Ghent , Belgium, 2015, https://ieeexplore.ieee.org/document/7177202

[10] MIT: “HRTF Measurements of a KEMAR Dummy-Head Microphone”
https://sound.media.mit.edu/resources/KEMAR.html

[11] ITA: “HRTF Datenbank”
https://www.akustik.rwth-aachen.de/cms/institut-fuer-hoertechnik-und-akustik/forschung/~lsly/hrtf-datenbank/

[12] Helmut Keller: “Digital State Variable Filters”
https://www.helmutkelleraudio.de/_downloads/bd50cee15835a4a58882d022a0ddc079

[13] Joep Vanlier: Download link for theYSFX plugin
https://github.com/JoepVanlier/ysfx/releases

[14] Helmut Keller: Download link for the author’s JSFX-plugins
https://www.helmutkelleraudio.de/_downloads/811aabeeac55bbaad0980342a67f65c4

[15] Helmut Keller: The authors website
https://www.helmutkelleraudio.de/